Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 29 (2012) 2649 – 2654

Procedia Engineering

www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

An Authentication and Key Agreement Mechanism for Multi-Domain Wireless Networks using Bilinear Pairings

Ming Luo[a*], Qi-jun Yan[b], Guo-qiang Jiang[c], Jian-feng Xu[a]

[a]*School of Software, Nanchang University, Nanchang 330047, China*
[b]*Network Department, Shenyang Institute of Education, Shenyang 110031, China*
[c]*Information Technology Center, China Mobile Group Liaoning Co.Ltd, Shenyang 110179, China*

Abstract

This paper presents an authentication and key agreement mechanism for multi-domain wireless networks using bilinear pairings. Based on the computational Diffie-Hellman assumption and the random oracle model, we show that the proposed scheme is secure against an uncertified user and a malicious registration server simultaneously. As compared with the recently proposed schemes, our scheme enjoys less computational cost and has higher security level by exploiting the certificateless public key cryptography system. Moreover, our scheme can be used to mutual authentication and key agreement between members of distinct domains where all the servers use different system parameters. Efficiency analysis of related the security and computation overheads are given to demonstrate that our scheme is well suited for mobile devices with limited computing capability.

Keywords: mutual authentication; key agreement; certificateless public key cryptography; bilinear pairing

1. Introduction

Now, handheld devices are popularly used by people and many mobile applications have been rapidly developed. Considering the limited computing capability of smart cards or mobile devices, the security scheme design based on traditional public-key systems is a nontrivial challenge because most cryptographic algorithms require many expensive computations. In 2006, Das *et al*. [1] proposed an efficient ID-based remote user authentication scheme with smart cards using bilinear pairings. Goriparthi *et al*. [2] showed that their scheme is insecure against forgery attack resulting in an adversary can always pass the authentication. Lately, Giri and Srivastava [3] proposed an improved scheme to withstand the

*Corresponding author.
E-mail address: lmhappy21@163.com.

1877-7058
doi:10.1016/j.proeng.2012.01.366

forgery attack. Unfortunately it was shown by Tseng *et al*. [4] that [3] has too expensive computational cost for smart cards with limited computing capability and is unable to be used for a multi-server environment and proposed a more efficient scheme. In 2010, Wu and Tseng [5] pointed out these above schemes do not provide mutual authentication and key agreement between the client and the server. Subsequently, Yoon and Yoo [6] analyzed the efficiency problem of the protocol [5] and then proposed a more efficient protocol that can reduce some hash operations and provide same security levels with an explicit key confirmation. However, all of schemes above face the key escrow issue as a result of adopting identity-based cryptography system. Moreover, their schemes assume that a single PKG will be responsible for issuing secret keys to members of a large-scale network or assume that different PKGs will share common system parameters.

In this paper, we propose a mutual authentication and key exchange mechanism for multi-domain wireless networks using bilinear pairings based on certificateless public key cryptography proposed by Al-Riyami and Paterson [7]. Currently, many CL-based cryptographic schemes such as signature schemes [8, 9] and authenticated key agreement protocols [10, 11] have been proposed for low-bandwidth channels and/or low-computation power. The smart card is a low power computing device while a server is regarded as a powerful node in the wireless networks. We shift the computational burden to the powerful node and reduce the computational cost required by smart cards. Compared with other secure schemes for wireless network regarding the security and computation overheads, we believe that our scheme is more efficient and more suitable for handheld devices with low computational capabilities on wireless communication. Our scheme has the following merits: (1) Users needn't submit their passwords to the registration server and they can freely choose and change their password without any assistance from the server; (2) The bilinear pairing operations to be computed only at the server side, and our scheme adopts CL-based short signatures to further induce the user computational cost. This makes our scheme especially attractive for the applications with a powerful server and number of handheld devices with low computational capabilities. (3) The scheme can be used to mutual authentication and key agreement between members of distinct domains using different system parameters. (4) The scheme is secure against an uncertified user and a malicious registration server simultaneously under the computational Diffie-Hellman assumption.

2. Proposed Scheme

In the following, we present our authentication and key agreement mechanism for multi-domain wireless networks using bilinear pairings based on certificateless public key cryptography. Unlike the scheme [3], in our proposed scheme each service server does not keep the system private keys to authenticate users. Users do not need to register with each service server individually and remember several identifiers and the corresponding secrets. Compared to the schemes [4,5,6], our scheme can be used to mutual authentication and key agreement between members of distinct domains using different system parameters, and our scheme is secure against an uncertified user and a malicious registration server simultaneously. The details of algorithms in the proposed scheme are given as follows:

Setup phase:

Suppose $G_{1\text{-}i}$ is an additive cyclic group of prime order q_i, and $G_{2\text{-}i}$ is a multiplicative cyclic group of the same order. We assume that solving CDHP is hard in group $G_{1\text{-}i}$. Suppose P_i is a generator of $G_{1\text{-}i}$. There exists a bilinear pairing map $\hat{e}_i$ from $G_{1\text{-}i}\times G_{1\text{-}i}$ to $G_{2\text{-}i}$ and cryptographic hash functions $H_{1\text{-}i}$: $\{0,1\}^n \rightarrow G_{1\text{-}i}$, $H_{2\text{-}i}$: $\{0,1\}^n \times G_{1\text{-}i} \times \{0,1\}^n \times G_{1\text{-}i} \rightarrow G_{1\text{-}i}$ and $H_{3\text{-}i}$: $G_{1\text{-}i} \rightarrow Z_{q_i}^*$. A server selects a random number $s_i \in Z_{q_i}^*$ as the private key and computes the public key $P_{pub\text{-}i} = s_i P_i$. Suppose RS obtains his private key s_1 and system public parameters are $<G_{1\text{-}1}, G_{2\text{-}1}, \hat{e}_1, q_1, P_1, P_{pub\text{-}1}, H_{1\text{-}1}, H_{2\text{-}1}, H_{3\text{-}1}>$, and SS chooses his private key s_2 and system public parameters are $<G_{1\text{-}2}, G_{2\text{-}2}, \hat{e}_2, q_2, P_2, P_{pub\text{-}2}, H_{1\text{-}2}, H_{2\text{-}2}, H_{3\text{-}2}>$.

Registration phase:

A user U first generates his username ID_U, then he submits his identity ID_U to the registration server RS for registration. The registration server RS computes $Q_U = H_{1\text{-}1}(ID_U)$ and uses his private key s_1 to

computes $D_U=s_1Q_U$. Finally, RS loads $\hat{e}_1$, P_1, $P_{pub\text{-}1}$, $H_{1\text{-}1}$, $H_{2\text{-}1}$, $H_{3\text{-}1}$, D_U, Q_U and ID_U into a smart card and issues the smart card to the user U. The server stores the ID_U into its database.

Mutual Authentication and Key Agreement phase:

This phase is executed whenever a user wants to log into the remote server to access the services. This phase is further divided into login, user authentication, server authentication and key agreement phases. In the login phase, user sends a login request to the SS. The SS first authenticates the user and then authenticates itself to the user. Finally, they establish a common session key after mutual authentication for the security of subsequent session message.

[Login phase]:

In the login phase, if the user U wants to access the SS with the identity ID_{SS}, U inserts his smart card into the terminal, for the first time, the smart card asks the user U to enter his password, U selects his password $s_U \in Z_{q_1}^*$, and then the smart card computes U's public key $PK_U=s_UP_1$, the smart card stores s_U and PK_U. Otherwise, the user enters his identity ID_U, his password and the service identity ID_{SS}. The smart card performs the following steps:

1. The smart card computes $Q'=H_{1\text{-}1}(ID_U)$ and $PK'=s_UP_1$, and then checks if $Q'=Q_U$ and $PK'=PK_U$. If they are correct, it continues next step, otherwise, terminates the operation

2. The smart card acquires the system public parameters of SS and the current time stamp T_1, then selects one random nonce $x \in Z_{q_1}^*$, computes $R_1=xP_2$, $W=H_{2\text{-}1}(ID_U, PK_U, T_1, R_1)$ and $V=D_U+s_UW$.

3. Finally, the smart card sends the login message $\sigma=(ID_U, ID_{SS}, T_1, R_1, V)$ to the service server SS, the login message can be viewed as a signature (R_1, V) on the message (ID_U, ID_{SS}, T_1).

[User Authentication Phase]:

As receives the login message (ID_U, T_1, R_1, V) at time T_2. The service server SS performs the following operations to verify the login message.

1. The SS first verifies the validity of ID_U and ID_{SS}, then verifies the time interval between T_2 and T_1. If $(T_2\text{-}T_1) \leqq \Delta t$, the SS proceeds to the next step. Otherwise, the login message is rejected. Here Δt denotes the expected valid time interval for transmission delay.

2. The SS computes $W=H_{2\text{-}1}(ID_U, PK_U, T_1, R_1)$ and accepts the login message if and only if the following equation holds: $\hat{e}_1(P_1,V)=\hat{e}_1(Q_U,P_{pub-1})\hat{e}_1(W,PK_U)$, otherwise the SS rejects it.

3. If the login message is correct, the SS acquires the current time stamp T_3 and selects one random nonce $y \in Z_{q_2}^*$, then computes $R_2=yP_1$, $KB_1=yPK_U$ and $Auth=H_{3\text{-}2}(KB_1)$. Finally, the SS sends $(R_2, T_3, Auth)$ to the user U.

[Server Authentication Phase]:

As receives the authentication message $(R_2, T_3, Auth)$ at time T_4. The user U verifies the validity of the time interval between T_3 and T_4 for transmission delay. If T_3 is valid, the user authenticates the service server SS by checking whether $Auth=H_{3\text{-}2}(KA_1)$, where $KA_1= s_UR_2$. It is obvious that $KA_1=s_UR_2=ys_UP_1=yPK_U=KB_1$.

[Key Agreement Phase]:

After mutual authentication between the user U and the service server SS, they respectively computes the session key $MK_A=H(K_{AB}||KA_1)$ and $MK_B=H(K_{AB}||KB_1)=MK_A=MK_{AB}$, where $K_{AB}=ID_U||PK_U||T_1||R_1||ID_{SS}||P_{pub\text{-}2}||T_3||R_2$ and H is a key derivation function. Thus, we come to the conclusion that the two communication entities successfully established a common session key MK_{AB}.

Password Change Phase:

This phase is invoked whenever the user U wants to change his password. This phase does not require any interaction with the servers and works as follows:

1. U inserts the smart card into the terminal and enters his identity ID_U and password s_U. The smart card computes $Q'=H_{1\text{-}1}(ID_U)$ and $PK'=s_UP_1$, and then checks if $Q'=Q_U$ and $PK'=PK_U$. If they are correct, it continues next step, otherwise, terminates the operation.

2. The smart card allows U to submits a new password s_U' , then the smart card computes $PK_U' = s_U' R$. The smart card stores new s_U' and PK_U' .

3. Security Analysis

For certificateless cryptosystems, the widely accepted notion of security was defined by Al-Riyami and Paterson in [7], there are two types of adversary with different capabilities:

Type I Adversary: This type of adversary A_1 models a dishonest user who does not have access to the master private key of registration server but has the ability to replace the public key of any entity with a value of his choice.

Type II Adversary: This type of adversary A_2 models a malicious registration server who has access to the master private key but cannot perform public keys replacement.

On one hand, we show that the service server can authenticate the user. In our scheme, the login messages (ID_U, T_1, R_1, V) is viewed as a signature (R_1, V) on the message (ID_U, ID_{SS}, T_1). For the adversary A_1, he can not forge the correct $R_1=xP_2$ and $V=D_U+s_UW$ to satisfy $\hat{e}_1(P_1,V)=\hat{e}_1(Q_U,P_{pub-1})\hat{e}_1(W,PK_U)$ without private keys pair of the user (s_U, D_U) under the assumption of CDHP. For the adversary A_2, he knows the other part of private key D_U of the user, but he without the user's other private key s_U, he can not compute s_UW under the assumption of CDHP. Hence, the adversary $A_{i(i=1,2)}$ cannot forge a valid signature on the message (ID_U, ID_{SS}, T_1) and the service server can authenticate the user.

On the other hand, we prove that the user U can authenticate the service server. In our scheme, after user authentication phase, the service server generates the authentication message $(R_2, T_3, Auth)$, the user can compute and verify the *Auth* value by running an instance of our authenticated key agreement protocol. We prove that adversary $A_{i(i=1,2)}$ in following Theorem 1 cannot compute the *Auth* value.

Theorem 1. The server authentication scheme is secure, provided that $H_{3\text{-}2}$, H_1 is random oracles and the Computational Diffie-Hellman problem is hard. Specifically, assume that adversary $A_{i(i=1,2)}$ has non-negligible advantage $Adv(A_i)$ in computing authentication value *Auth*, making at most q_c Create-User queries and q_p Password-Extract queries. Let q_n be the total number of the oracles that A_i creates. Then there exists an algorithm C solve the CDH problem with an advantage $2(q_c - q_p)Adv(A_i)/(q_c \cdot q_n)$.

Proof. We assume the simulator C receives a random instance (P, aP, bP) of the Computational Diffie-Hellman problem. His goal is to compute abP. C will run A_i as a subroutine to solve the CDH problem with non-negligible probability. To maintain consistency between queries made by A_i, C keeps the following lists: L_1 for query/response pairs to random oracle H_1; L_u of the queries made by A_i to the Create-User oracle and L_h of some of the queries made by A_i to the $H_{3\text{-}2}$ oracle. At the beginning of the game, C gives A_i the system parameters of RS and SS, and gives A_2 the private key s of RS, we define RS's system public parameters are $<G_1,G_2,\hat{e},q,P,P_{pub},H_1,H_2,H_3>$ and SS's system public parameters are $<G_{1\text{-}2},G_{2\text{-}2},\hat{e}_2,q_2,P_2,P_{pub\text{-}2},H_{1\text{-}2},H_{2\text{-}2},H_{3\text{-}2}>$

The algorithm C selects one random integers τ from $\{1,2, ..., q_n\}$ and works by interacting with A_i as follows, where A_2 doesn't need to access Private-Key-Extract and Public-Key-Replace oracles:

Create-User: C chooses one random numbers $i_b \in \{1, 2, ..., q_c\}$ first. At the i_b-th query, C sets $s_U=\perp$, $ID_b=ID_U$ and $PK_U=aP$. For others queries, C chooses a random number $s_U \in Z_q^*$ and computes $PK_U= s_UP$. In both cases, C adds (ID_U, s_U, PK_U) into the list L_u and returns PK_U to A_i.

***H_1* queries:** C chooses a random number $w \in Z_q^*$, and sets $H_1(ID_U)=wP$, then C will puts the pair $(ID_U, w, H_1(ID_U))$ in list L_1 and answers $H_1(ID_U)$.

***$H_{3\text{-}2}$* queries:** Upon receiving a $H_{3\text{-}2}$ query, C first searches L_h for the tuple with (K_1, h), where $K_1 \in G_1$. If the requested input is already on the list, then the corresponding h is returned, otherwise a random $h \in \{0,1\}^n$ is responded and a new entry is inserted into the list L_h.

Public-Key-Replace: C replaces the original public key PK_U with PK_U' if ID_U has been created. Otherwise, C executes Create-User query to generate (ID_U, s_U, PK_U), then sets $PK_U = PK_U'$ and adds (ID_U, s_U, PK_U') to the L_u. Here, to replace a public key, the password value corresponding to the new public key is not required.

Password-Extract: On a Password-Extract query of ID_U, We assume that Create-User query for ID_U has been asked. If $ID_U = ID_b$, then C fails and stops. Otherwise, C searches a pair (ID_U, s_U, PK_U) corresponding to ID_U in the list L_u, then return s_U to A_i.

Send queries: For any oracle $\Pi^n_{A,SS}$, at the τ-th Send query, C answers by $R_2=bP$. For others queries, C chooses a random number $d_i \in Z_q^*$ and answers d_iP.

Reveal queries: Upon receiving a Reveal query, C outputs the appropriate session key, except if A_i asks the oracle $\Pi^\tau_{A,SS}$ to ask the Test query, then C aborts.

Test queries: At some point in the simulation, A_i will ask a single Test query of some oracle. If A_i does not choose the guessed oracle $\Pi^\tau_{A,SS}$ to ask the Test query, then C aborts; otherwise, C randomly picks a value β from the session key space and responds to A_i with β.

Output: At the end of the game, the algorithm A_i outputs its guess.

Solving the CDH Problem: C picks a tuple of the form (K_1, h) from L_h and returns K_1 as the response to the CDH challenge.

Now we evaluate the probability that C does not abort, Note that C fails if A_i has asked a Private-Key-Extract query on ID_b. We know that the probability for C not to fail is $(q_c-q_p)/q_c$; Further, if the test session is the τ-th oracle, then the simulation goes through. The probability that the simulator has chosen the right session is $1/q_n$, because a randomly chosen oracle is the initiator of the test session is $1/q_n$. We have: $Adv(C \text{ does not abort}) > (q_c-q_p)/q_c \cdot 1/q_n = (q_c-q_p)/(q_c \cdot q_n)$

According to the simulation of the Send query, the test oracle $\Pi^\tau_{A,SS}$ must have obtained the value $R_2=bP$. The oracle should hold an authentication value $Auth$ of the form $H_{3\text{-}2}(K_1)$, in which $K_1=abP$.

Let $\hat{H}$ be the event that abP as K_1 has been queried to $H_{3\text{-}2}$. Because $H_{3\text{-}2}$ is a random oracle, we have $P[A_i \text{ wins}|\neg\hat{H}] = 1/2$. Then

$P[A_i \text{ wins}] = P[A_i \text{ wins}|\neg\hat{H}]P[\neg\hat{H}] + P[A_i \text{ wins}|\hat{H}]P[\hat{H}] \leq P[A_i \text{ wins}|\neg\hat{H}]P[\neg\hat{H}] + P[\hat{H}] = 1/2 + 1/2(P[\hat{H}])$

It follows that $P[\hat{H}] \geq 2Adv(A_i)$.Combining all the above results, we have that C solves the CDH problem with probability at least $2(q_c - q_p)Adv(A_i)/(q_c \cdot q_n)$, contradicting to the hardness of the CDH problem.

4. Protocol Comparison

In this section, we compare the efficiency of our scheme with Yoon and Yoo's scheme [6] regarding the security and computation overheads not including precomputation overheads. We use notations *mul*, *add*, *bp* and *h* as abbreviations for multiplication in G_1, addition in G_1, bilinear pairing operation and one-way hash function operation respectively.

As shown in the Table 1, both schemes do not require expensive bilinear pairing operation on the client side, which makes them more efficient than others schemes [1, 3]. Compared with the Yoon and Yoo's scheme, our scheme enjoys less operation cost. Moreover, our scheme can be used to mutual authentication and key agreement between members of distinct domains using different system parameters, and our scheme is secure against an uncertified user and a malicious registration server simultaneously. Hence, consider the wireless user with limited computing capability and communication security it may be that our authentication and key agreement scheme is more applicable.

Table 1. A comparison of efficiency

	Yoon and Yoo's scheme [6]	Our scheme

Client	4*mul*+*add*+2*h*	3*mul*+*add*+3*h*
Server	2*mul*+*add*+2*bp*+2*h*	2*mul*+2*bp*+3*h*

5. Conclusion

In this paper, we have proposed an authentication and key agreement mechanism for multi-domain wireless networks using bilinear pairings based on certificateless public key cryptography. We have shown that the proposed scheme is secure against an uncertified user and a malicious registration server simultaneously under the computational Diffie-Hellman assumption in the random oracle. By exploiting the certificateless public key cryptography system, our scheme successfully eliminates the key escrow issue which is inherent in identity-based cryptography. In the proposed scheme, we shift the computational burden to the server; moreover, our scheme adopts CL-based short signatures to further induce the user computational cost. As a result, the computational cost required by the user is reduced to be well suited for smart cards. As compared with the recently proposed schemes, our scheme has better performance in term of the security and computation overheads.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under contract no. 61070139 and the Science and Technology Foundation of the Education Department of Jiangxi Province under grant no. GJJ11039.

References

[1] Das ML, Saxena A, Gulati VP, et al. A novel remote user authentication scheme using bilinear pairings. Computers and Security 2006; 25(3): 184–189.

[2] Goriparthi T, Das ML, Negi A, et al. Cryptanalysis of recently proposed Remote User Authentication Schemes. Technical Report 028, Cryptology ePrint Archive, 2006.

[3] Giri D, Srivastava PD. An improved remote user authentication scheme with smart cards using bilinear pairings. Technical Report 274, Cryptology ePrint Archive, 2006.

[4] Tseng YM, Wu TY, Wu JD. A Pairing-Based User Authentication Scheme for Wireless Clients with Smart Cards. Informatica 2008; 19(2): 285-302.

[5] Wu TY, Tseng YM. An efficient client authentication and key exchange protocol for mobile client–server environment. Computer Networks 2010; 54: 1520–1530.

[6] Yoon EJ, Yoo KY. A New Efficient ID-based User Authentication and Key Exchange Protocol for Mobile Client-Server Environment, Proc. ICWITS 2010, 1-4.

[7] Al-Riyami SS, Paterson KG. Certificateless Public Key Cryptography. Proc. Cryptography-ASIACRYPT 2003, 452–473.

[8] Ma CB, Ao J. Certificateless Group Oriented Signature Secure Against Key Replacement Attack. International Journal of Network Security 2011; 12(1): 1-6.

[9] Choi KY, Park JH, Lee DH. A new provably secure certificateless short signature scheme. Computers & Mathematics with Applications 2011; 61(7): 1760-1768.

[10] He DB, Chen YT, Chen JH, etal. A new two-round certificateless authenticated key agreement protocol without bilinear pairings. Mathematical and Computer Modelling 2011; 54: 3143-3152.

[11] Xiong H, Wu QH, Chen Z. Toward Pairing-Free Certificateless Authenticated Key Exchanges. Proc. ISC 2011, 79-94.